ABSTRACT OF THE DISCLOSURE

Factored assert chains allow for improved tracking of implicit information in a computer program. The compiler may generate assert statements at various points in the program where there is implicit information. The dominator tree for the program or section of program may then be constructed. Then ϕ-nodes may be inserted throughout a control flow graph. Following that, for each statement in the program or section of program, an assert chain may be constructed from each use to the most recent available assert statement for the variable. Then, if the statement is an assert statement, each use may be kept track of as a mapping to an assertion, otherwise a mapping of any reference to an assert statement for each definition may be deleted. This may then iterate through the dominator tree. At the end, a series of factored assert chains remains, which may be utilized by the compiler to improve the efficiency of generated code.